RACAL INSTRUMENTS™

VXI MODULAR INSTRUMENTATION

2460 SERIES

Publication No. 980833 Rev. A

Astronics Test Systems Inc.
4 Goodyear, Irvine, CA 92618
Tel: (800) 722-2528, (949) 859-8999; Fax: (949) 859-7139

atsinfo@astronics.com atssales@astronics.com
atshelpdesk@astronics.com http://www.astronicstestsystems.com

Copyright 1998 by Astronics Test Systems Inc. Printed in the United States of America. All rights reserved. This book or parts thereof may not be reproduced in any form without written permission of the publisher.

THANK YOU FOR PURCHASING THIS ASTRONICS TEST SYSTEMS PRODUCT

For this product, or any other Astronics Test Systems product that incorporates software drivers, you may access our web site to verify and/or download the latest driver versions. The web address for driver downloads is:

http://www.astronicstestsystems.com/support/downloads

If you have any questions about software driver downloads or our privacy policy, please contact us at:

atsinfo@astronics.com

WARRANTY STATEMENT

All Astronics Test Systems products are designed to exacting standards and manufactured in full compliance to our AS9100 Quality Management System processes.

This warranty does not apply to defects resulting from any modification(s) of any product or part without Astronics Test Systems express written consent, or misuse of any product or part. The warranty also does not apply to fuses, software, non-rechargeable batteries, damage from battery leakage, or problems arising from normal wear, such as mechanical relay life, or failure to follow instructions.

This warranty is in lieu of all other warranties, expressed or implied, including any implied warranty of merchantability or fitness for a particular use. The remedies provided herein are buyer's sole and exclusive remedies.

For the specific terms of your standard warranty, contact Customer Support. Please have the following information available to facilitate service.

1. Product serial number
2. Product model number
3. Your company and contact information

You may contact Customer Support by:

E-Mail:	atshelpdesk@astronics.com	
Telephone:	+1 800 722 3262	(USA)
Fax:	+1 949 859 7139	(USA)

RETURN OF PRODUCT

Authorization is required from Astronics Test Systems before you send us your product or sub-assembly for service or calibration. Call or contact Customer Support at 1-800-722-3262 or 1-949-859-8999 or via fax at 1-949-859-7139. We can also be reached at: atshelpdesk@astronics.com.

If the original packing material is unavailable, ship the product or sub-assembly in an ESD shielding bag and use appropriate packing materials to surround and protect the product.

PROPRIETARY NOTICE

This document and the technical data herein disclosed, are proprietary to Astronics Test Systems, and shall not, without express written permission of Astronics Test Systems, be used in whole or in part to solicit quotations from a competitive source or used for manufacture by anyone other than Astronics Test Systems. The information herein has been developed at private expense, and may only be used for operation and maintenance reference purposes or for purposes of engineering evaluation and incorporation into technical specifications and other documents which specify procurement of products from Astronics Test Systems.

TRADEMARKS AND SERVICE MARKS

All trademarks and service marks used in this document are the property of their respective owners.

- Racal Instruments, Talon Instruments, Trig-Tek, ActivATE, Adapt-A-Switch, N-GEN, and PAWS are trademarks of Astronics Test Systems in the United States.

DISCLAIMER

Buyer acknowledges and agrees that it is responsible for the operation of the goods purchased and should ensure that they are used properly and in accordance with this document and any other instructions provided by Seller. Astronics Test Systems products are not specifically designed, manufactured or intended to be used as parts, assemblies or components in planning, construction, maintenance or operation of a nuclear facility, or in life support or safety critical applications in which the failure of the Astronics Test Systems product could create a situation where personal injury or death could occur. Should Buyer purchase Astronics Test Systems product for such unintended application, Buyer shall indemnify and hold Astronics Test Systems, its officers, employees, subsidiaries, affiliates and distributors harmless against all claims arising out of a claim for personal injury or death associated with such unintended use.

FOR YOUR SAFETY

Before undertaking any troubleshooting, maintenance or exploratory procedure, read carefully the **WARNINGS** and **CAUTION** notices.

This equipment contains voltage hazardous to human life and safety, and is capable of inflicting personal injury.

If this instrument is to be powered from the AC line (mains) through an autotransformer, ensure the common connector is connected to the neutral (earth pole) of the power supply.

Before operating the unit, ensure the conductor (green wire) is connected to the ground (earth) conductor of the power outlet. Do not use a two-conductor extension cord or a three-prong/two-prong adapter. This will defeat the protective feature of the third conductor in the power cord.

CAUTION
SENSITIVE ELECTRONIC DEVICES
DO NOT SHIP OR STORE NEAR STRONG ELECTROSTATIC, ELECTROMAGNETIC, MAGNETIC OR RADIOACTIVE FIELDS

Maintenance and calibration procedures sometimes call for operation of the unit with power applied and protective covers removed. Read the procedures and heed warnings to avoid “live” circuit points.

Before operating this instrument:

1. Ensure the proper fuse is in place for the power source to operate.

2. Ensure all other devices connected to or in proximity to this instrument are properly grounded or connected to the protective third-wire earth ground.

If the instrument:

- fails to operate satisfactorily
- shows visible damage
- has been stored under unfavorable conditions
- has sustained stress

Do not operate until performance is checked by qualified personnel.

DECLARATION OF CONFORMITY

Ser. No. CE 021

We:

Supplier's Name: Racal Instruments Ltd.

Supplier's Address: 480 Bath Road
Slough
Berks.
SL1 6BE

declare under our sole responsibility that the following product(s):

Product Name: 2460 Series VXI module

Product Model Number(s): 2461-Cx-Cx (where x=a to f)

Product Options: Not Applicable

to which this declaration relates conform(s) with the following standard(s) or other normative document(s):

Safety: EN 61010-1:1993/A2:1995

EMC:

Emissions: EN55022:1988, Class B/BSEN 55022:1995, Class B

Immunity: EN50082-1:1992
IEC 801-2:1991/BSEN 60801-2:1993, 4kV CD, 8kV AD, performance criteria b).
IEC 801-3:1984/IEC 1000-4-3:1995, 3V/m, performance criteria b).
IEC 801-4:1988/BSEN 61000-4-4:1995, 1kV power lines, 500V signal & data lines, performance criteria b).
IEC 1000-4-5:1995, 1kV L/N-E, 500V L-N, performance criteria b).
IEC1000-4-6:1996, 3V, performance criteria b).
IEC 1000-4-11:1994, duration of dip: 1 cycle, performance criteria c).

Supplementary Information:

The products herewith comply with the requirements of the Low Voltage Directive 73/23/EEC (as amended by Directive 93/68/EEC) and the EMC Directive 89/336/EEC (as amended by Directives 92/31/EEC and 93/68/EEC) and carry the CE mark accordingly. The products were tested in a typical configuration.

Name of Authorised Signatory: (PRINT) C M ROBERTSON

Professional Title of Signatory: (PRINT) M.D.

Signature of Authorised Signatory: 10JUL98

Place and Date of Issue: Slough, Berks., England (town,county,country) _____ (date)

RIL 071D

This page was left intentionally blank.

Table of Contents

Chapter 1 1-1

GENERAL 1-1

The 2460 Series VXIbus Module 1-1

Modularity and Flexibility 1-2

Cost Effective 1-2

Field Upgradable 1-2

Chapter 2 2-1

INSTALLATION INSTRUCTIONS 2-1

Unpacking and Inspection 2-1

Reshipment Instructions 2-1

Installation 2-2

The VXI*plug&play* Driver and Product Manuals 2-4

Checking To See That The Module is Operational 2-4

Removing the Module From The Mainframe 2-4

Recommended Controllers 2-4

Chapter 3 3-1

OPERATION OF THE 2461 3-1

VIA THE VXI*plug&play* DRIVER 3-1

VXI*plug&play* Driver 3-1

Using The Soft Front Panel 3-1

Chapter 4 4-1

PROGRAMMING THE 2461 MODULE 4-1

Introduction 4-1

General Principles 4-1

Operation Via The SCPI Command Set 4-2

SCPI Principles 4-2

Sending and Receiving SCPI Commands Over the VXI*bus* 4-4

Selecting an Application 4-4

Status and Error Reporting 4-5

System Timeout 4-5

General Commands 4-6

List of General SCPI Commands 4-6

INSTrument Subsystem 4-7

STATus subsystem 4-9

SYSTem subsystem 4-13

Common commands 4-16

Error Messages 4-25

No Errors 4-25

Command Errors 4-25

Execution Errors 4-27

Instrument-Specific Errors 4-28

VXI*bus* Word Serial commands 4-29

Abort Normal Operation 4-29

Assign Interrupt Line 4-30

Asynchronous Control Mode 4-31

Begin Normal Operation 4-32

Byte Available 4-32
Byte Request 4-33
Clear 4-33
Control Event 4-34
End Normal Operation 4-34
Read Interrupt Line 4-35
Read Interrupts 4-35
Read Protocol 4-36
Read Protocol Error 4-37
Read STB 4-38

List of Figures

Figure 1-1, The 2460 Series VXIbus Module 1-1

Figure 1-2, The 2460 VXIbus Module 1-2

Figure 2-1, The installation of the 2460 Series VXIbus Module 2-2

Figure 2-2, Location of S1 2-3

Figure 3-1, Main Soft Front Panel of the 2461 application 3-2

Figure 3-2, ‘System Functions’ secondary Panel of the 2461 application 3-3

Figure 3-3, Status 3-4

Figure 3-4, Self Test 3-4

Figure 3-5, ‘About’ secondary Soft Front Panel of the 2461 application 3-5

Figure 4-1, Operation Status Register. 4-10

Figure 4-2, Status Byte and Service Request 4-17

Figure 4-3, Standard Event Status 4-18

DOCUMENT CHANGE HISTORY

Revision	Date	Description of Change
A	11/18/08	Revised per EO 29469 Revised format to current standards. Company name revised throughout manual. Manual now revision letter controlled. Added Document Change History Page v.

This page was left intentionally blank.

Chapter 1

GENERAL

The 2460 Series VXIbus Module

The 2460 series is based on an innovative module produced by Astronics Test Systems. Its design was developed to reduce the size and cost of test systems by allowing more than one function in a single VXIbus module.

Figure 1-1, The 2460 Series VXIbus Module

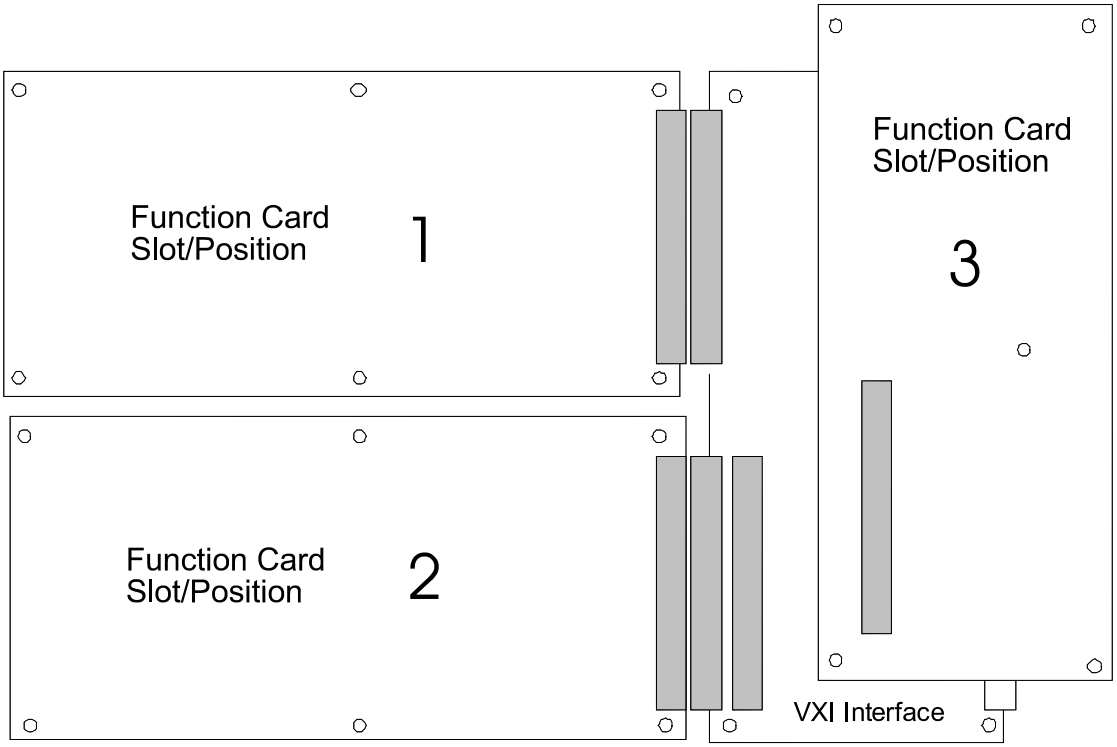


Figure 1-2, The 2460 VXIbus Module

Modularity and Flexibility

The 2460 Series VXIbus Module will allow up to 3 separate function cards to be fitted in a single “C” size slot. Slot or positions 1 and 2 provide access to the front panel, while slot/position 3 allows for future development or enhancements.

Cost Effective

In some applications, this will allow the user to expand or reduce the number of modules/cards with in an existing system.

Additionally fewer modules within a VXI chassis will allow a user to choose a six slot chassis instead of a standard thirteen slot chassis.

Field Upgradable

2460 Series are field upgradable. The function cards are automatically recognized when installed by the firmware.

Chapter 2

INSTALLATION INSTRUCTIONS

Unpacking and Inspection

1. Remove the 2460 Series module and inspect it for damage. If any damage is apparent, inform the carrier immediately. Retain shipping carton and packing material for the carrier's inspection.

2. Verify that the pieces in the package you received contain the correct 2460 Series module option and the 2460 Series Users Manual. Notify Customer Support if the module appears damaged in any way. Do not attempt to install a damaged module into a VXI chassis.

3. The 2460 Series module is shipped in an anti-static bag to prevent electrostatic damage to the module. Do not remove the module from the anti-static bag unless it is in a static-controlled area.

Reshipment Instructions

1. Use the original packing when returning the switching module to Astronics Test Systems for calibration or servicing. The original shipping carton and the instrument's plastic foam will provide the necessary support for safe reshipment.

2. If the original packing material is unavailable, wrap the switching module in an ESD Shielding bag and use plastic spray foam to surround and protect the instrument.

3. Reship in either the original or a new shipping carton.

Installation

The installation of the 2460 Series VXIbus module into a chassis or mainframe is shown in the following diagram:

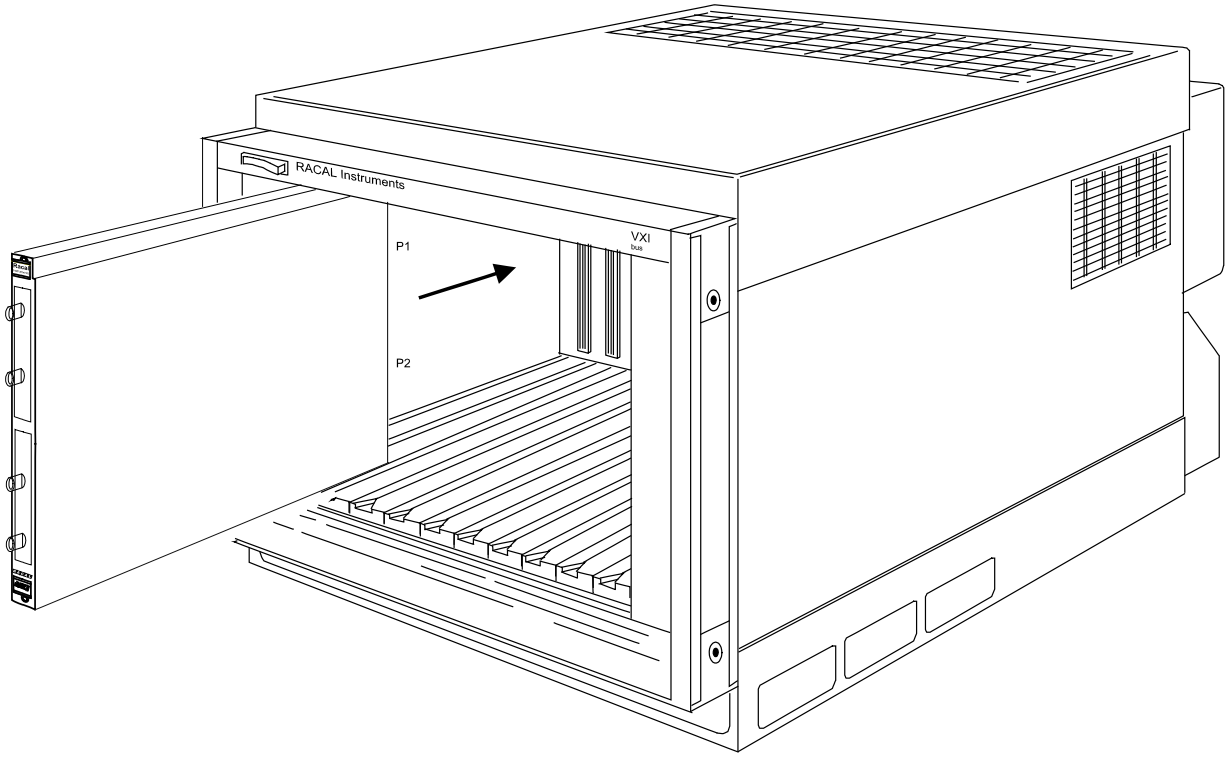


Figure 2-1, The installation of the 2460 Series VXIbus Module

Before commencing installation of the 2460 Series VXI Module instrument, carry out a visual inspection of it. Pay particular attention to connectors P1 and P2 on the rear of the instrument, and ensure that there are no bent, damaged or missing pins, any damage to the instrument should be repaired before proceeding.

Refer to the product identification label on the side of the module for system integration information relating to voltage, power, and cooling requirements to be supplied by the VXI chassis.

Switch, S1, is used to set the Logical Address of the module to a static address or allow the controller to dynamically allocate an address. The switch is marked for least significant bit (LSB) and MSB (most significant bit) and logic states 1 and 0.

Static Logical Addresses may be set between the values 1 and 254. If S1 is set to 255, then the controller dynamically allocates the next available Logical Address. The factory default is 254.

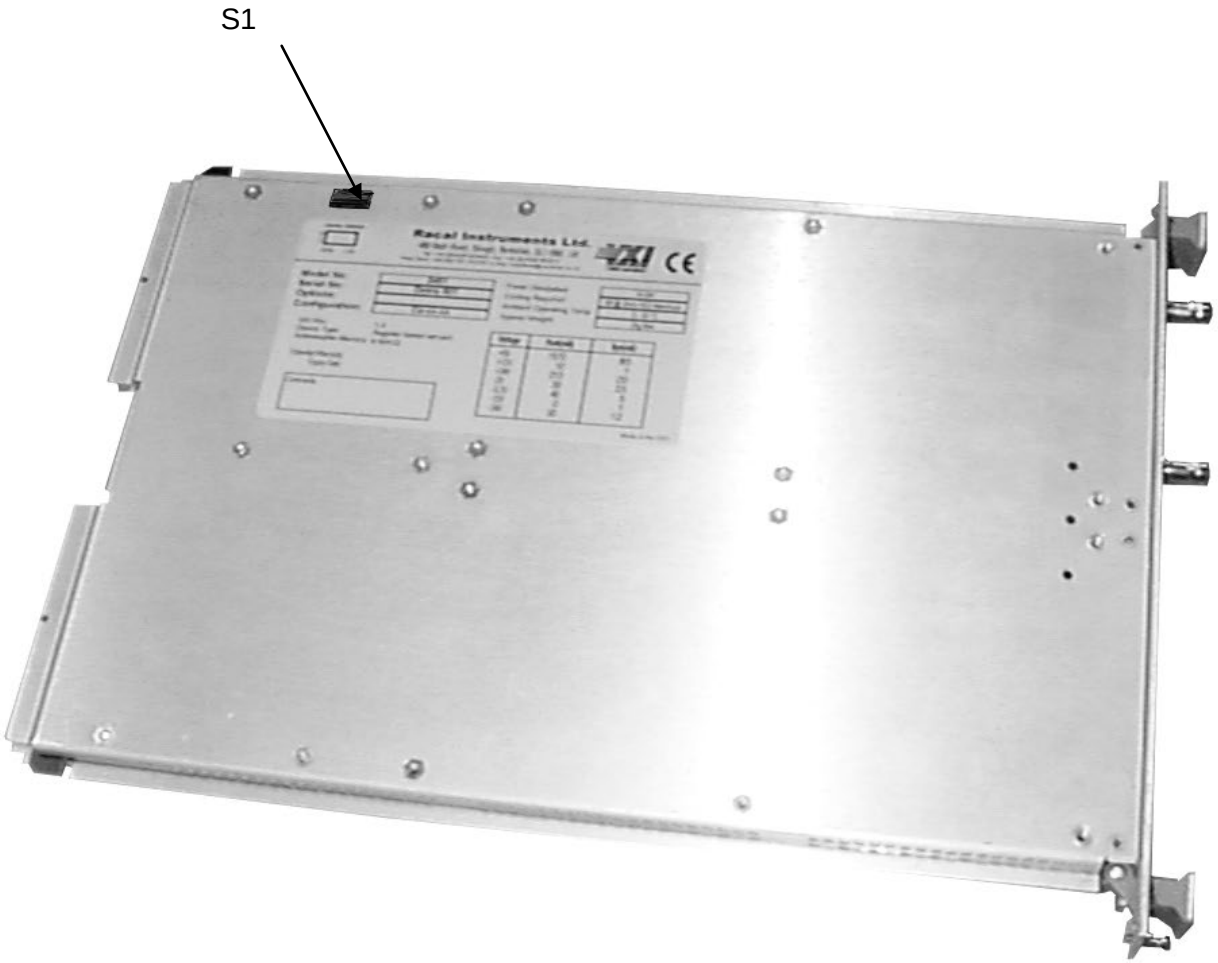


Figure 2-2, Location of S1

To install the 2460 Series VXI Module in a C size VXI mainframe, ensure that the it has the rear connector P1 oriented to mate with the corresponding connector on the mainframe backplane. Align the module with the guides for the slot selected, and slide the module into the mainframe. Push the module home to connect it to the mainframe. Secure the module to the mainframe with the captive screws provided at the top and bottom edges of the front panel.

Poor mechanical alignment of the module's rear connectors, P1 and P2, may require that it be reseated in the VXI mainframe. Do NOT use undue force to seat the module.

The VXI*plug&play* Driver and Product Manuals

CD part number 980833-003 supplied with this unit contains the 2460 series driver. To install the VXI*plug&play* driver, the computer must be running WIN 95 or WIN NT.

NOTE: This CD also contains the 2460 Series manual part number 980833 and the 2461C Universal Counter Timer manual 980833-002.

Checking To See That The Module is Operational

Power-up the chassis/mainframe. Ensure that the SYSFAIL LED on the module's front panel illuminates and then extinguishes after a few seconds. This indicates that the module has passed its power-up self test, and is now ready for use.

1. Initialize the system as appropriate (eg. for a PCI MXI-2 system) run Resman

2. Run the Self-Test from the VXI*plug&play* soft front panel and check that a PASS is indicated.

Removing the Module From The Mainframe

To remove the module from the VXI mainframe, power-down the mainframe, and release the captive screws that secure the module to the mainframe. Use the plastic levers provided on the top and bottom edges to eject the module from the mainframe. Pull the module along the guides provided and out of the mainframe.

Recommended Controllers

To utilize the full capability of the module, the slot 0 controller should have A32 addressing capability. A controller with only A16/A24 capability may be used but the speed advantage gained with shared memory access will not be realized. Note: the module has no A24 capability.

Chapter 3
OPERATION OF THE 2461 VIA THE VXI*plug&play* DRIVER

VXI*plug&play* Driver

Use of the VXIplug&play driver allows the operator to develop programs without intimate knowledge of the command set, SCPI or IEEE 488.2 standards.

The driver consists of a number of functions, help files, soft front panels, knowledge base and help facility. The front panels give the programmer an easy conversion of front panel settings into function calls. Alternately, the functions calls can be written directly by the programmer. The soft front panel is an application that allows interactive control of the 2461 using the driver functions.

The driver consists of a series of functions, some of which are mandatory in the VXIplug&play standard. The remainder of the functions are device defined.

To use the driver it is first necessary to first to establish a link to the 2461. This is done by the ri2461_init function. This function will return a handle for future access to the 2461 plugged into the VXI system. All other functions will require this handle as a passed parameter. If more than one 2461 exists in a VXI system then this function must be called for each instrument using the instrument descriptor to differentiate between instruments. The instrument descriptor is generated by the resource manager. A device defined function, ri2461_autoFind, is provided to determine the number of 2461 instruments together with their descriptors and slot numbers.

Using The Soft Front Panel

The soft front panel (shown in **Figure 3-1**) gives the operator interactive control over the 2461 allowing the user to become more familiar with instrument operation. All major functions are provided.

To use the soft front panel the driver must have been installed as previously described. The soft front panel application should be launched. The soft front panel application will now search for a 2461, or multiple 2461, in the VXI system. If a window appears with the message ‘No 2461 found in Chassis. Do you want to run in Demo mode?’ and a 2461 is plugged in then the

troubleshooting guide in the help file should be followed.

If it is desired to run the demonstration version then the 'YES' option should be selected.

If one 2461 is found then the main front panel of the 2461 will appear on the screen and appear similar to that shown in **Figure 3-1**. If more that one 2461 is found then a selection list will be presented, enabling the user to choose which one to operate. Then the main front panel (**Figure 3-1**) will appear for that 2461.

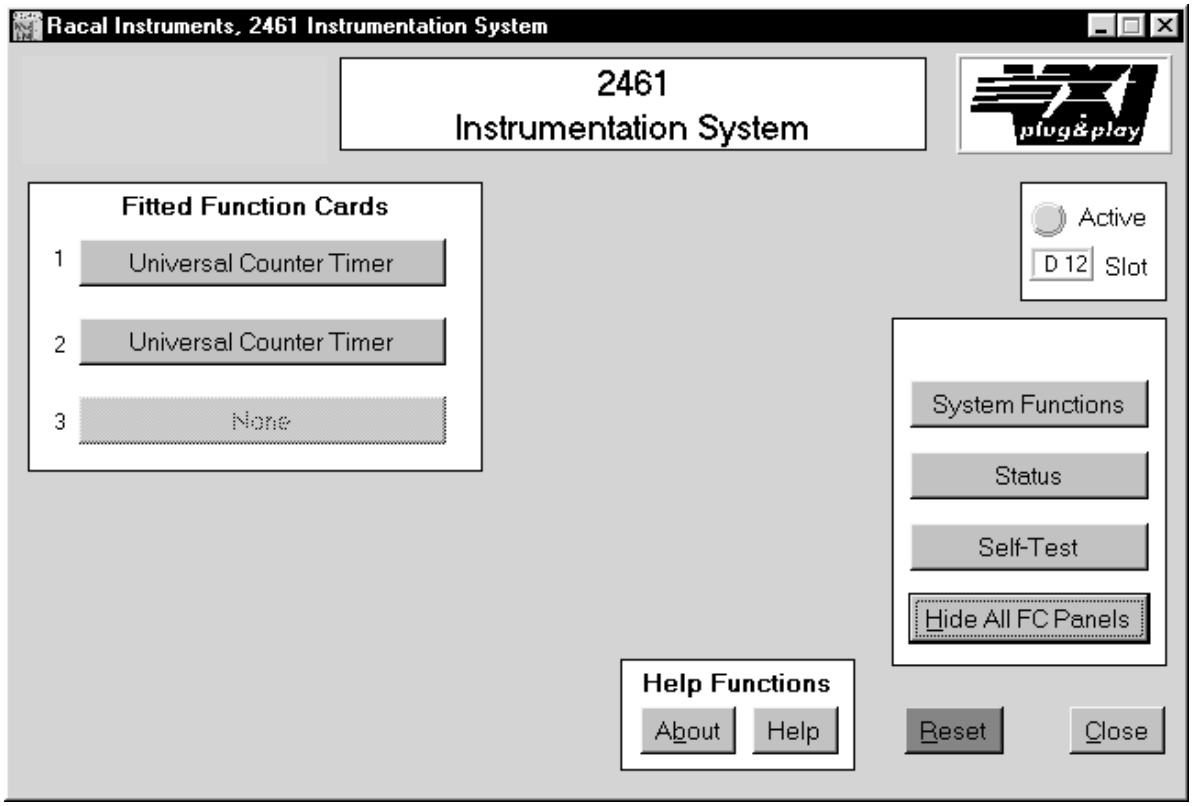


Figure 3-1, Main Soft Front Panel of the 2461 application

All installed function cards are shown in selectable buttons. Clicking on one of these will bring up the main front panel of that function card. These function card sub-panels may contain further sub panels depending on the application.

The ***System Functions*** panel, shown in **Figure 3-2**, enables the user to set the system time-out, shows the most recent system error and the SCPI specification revision to which the instrument conforms.

The ***Status*** button displays the status byte of the 2461.

The ***Self-Test*** button initiates a self-test on the module and

returns the result as a text message. Note that the self-test checks all function cards installed in the 2461 module.

The ***Hide All FC Panels*** button temporarily disables the active secondary panels.

The ***About*** button, shown in **Figure 3-3**, contains information on the instrument driver and firmware of the 2461 and function cards.

The ***Help*** button supports on-line help which provides the user with an efficient way to use the panel without referring to printed documentation. Note that clicking on a control with the right mouse button will provide help for that control.

The ***Reset*** button re-initializes the module and all the function cards fitted to their default conditions.

The ***Close*** button releases the interface and stops the execution of the soft front panel application.

The Active LED indicates the status. A Green LED indicates a working 2461 or an LED in the OFF state and a D in front of the slot number indicates the soft front panel is in DEMO mode.

The slot number indicates the physical position of the selected 2461.

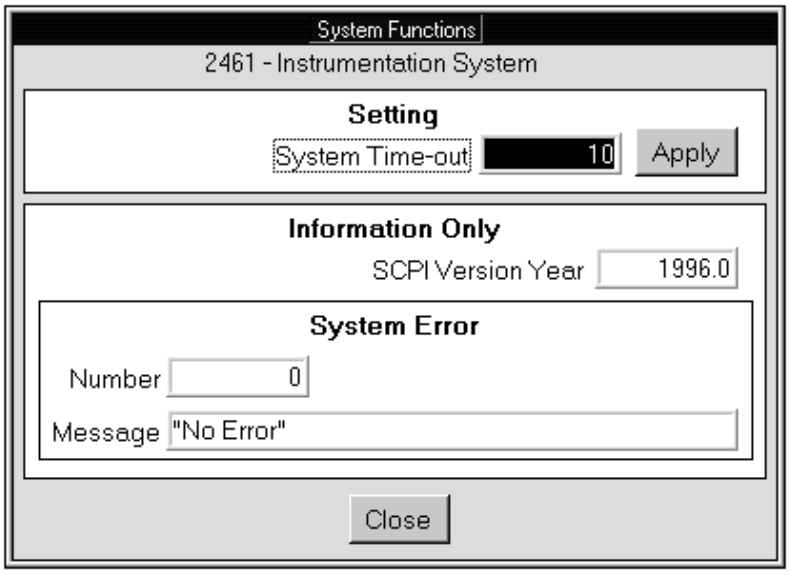


Figure 3-2, 'System Functions' secondary Panel of the 2461 application

The ***Setting*** displays the System Time-out. The Time-out setting specifies the maximum time a query such as MEAS? or READ? can take before an error is generated. A value of 0 disables the time-out.

Information Only displays the SCPI version year, specifies the SCPI version to which the instrument conforms.

The ***System Error*** displays Error Number. It specifies the error code of the most recent E*rror Message* generated by the 2461. If more than one error is present in the queue a Next control may be operated successively until no more errors are available.

The ***Apply*** command button initiates the change to the Time-out setting.

The ***Close*** command button closes the panel.

The Status window displays the status byte, refer to Chapter 4, page 4-17, **Figure 4-2**.

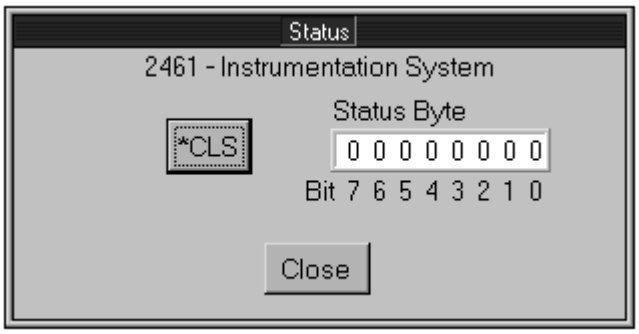


Figure 3-3, Status

The Self-test returns the result as a text message. Note that the self-test checks all function cards installed in the 2461 module.

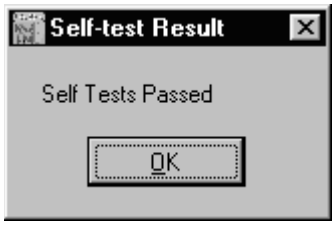


Figure 3-4, Self Test

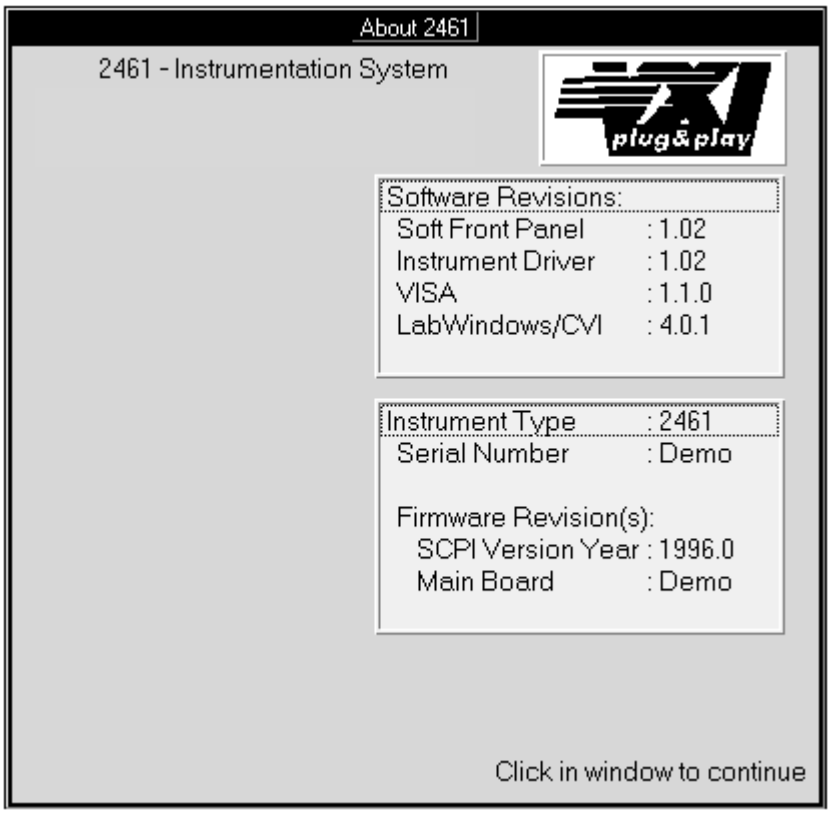


Figure 3-5, 'About' secondary Soft Front Panel of the 2461 application

This panel shows the Software revision of the soft front panel, instrument driver, VISA library and LabWindows/CVI. As well as instrument details such as, its type and serial number, the SCPI revision to which the software complies, the firmware revision of the main board and any function card(s).

This page was left intentionally blank.

Chapter 4

PROGRAMMING THE 2461 MODULE

Introduction

This Section provides information about using the general SCPI commands, and how to establish the correct system configurations for the VXI application function cards mounted in the 2461 VXI Module.

The following information is provided:

A general description of SCPI Commands and Syntax Rules.

General Commands and Common Commands available.

Common Error Messages.

General Principles

The 2461 can be operated in two ways, using the supplied Plug and Play driver or, by sending and receiving SCPI compliant commands and data. If the Plug and Play driver is used, the user has the option of running the soft front panel application to get interactive control.

The 2461 is a VXI*bus* Module with up to three function cards plugged in at any one time. Since all function cards by definition are accessed by the same logical address a means of accessing each application is provided.

The 2461 module processor has a general command set which is valid at all times. This general command set contains commands to select an application. Each application has its own command set. Once an application has been selected then the command set for that application will become valid. Note that a particular command may be valid for more than one application.

Operation Via The SCPI Command Set

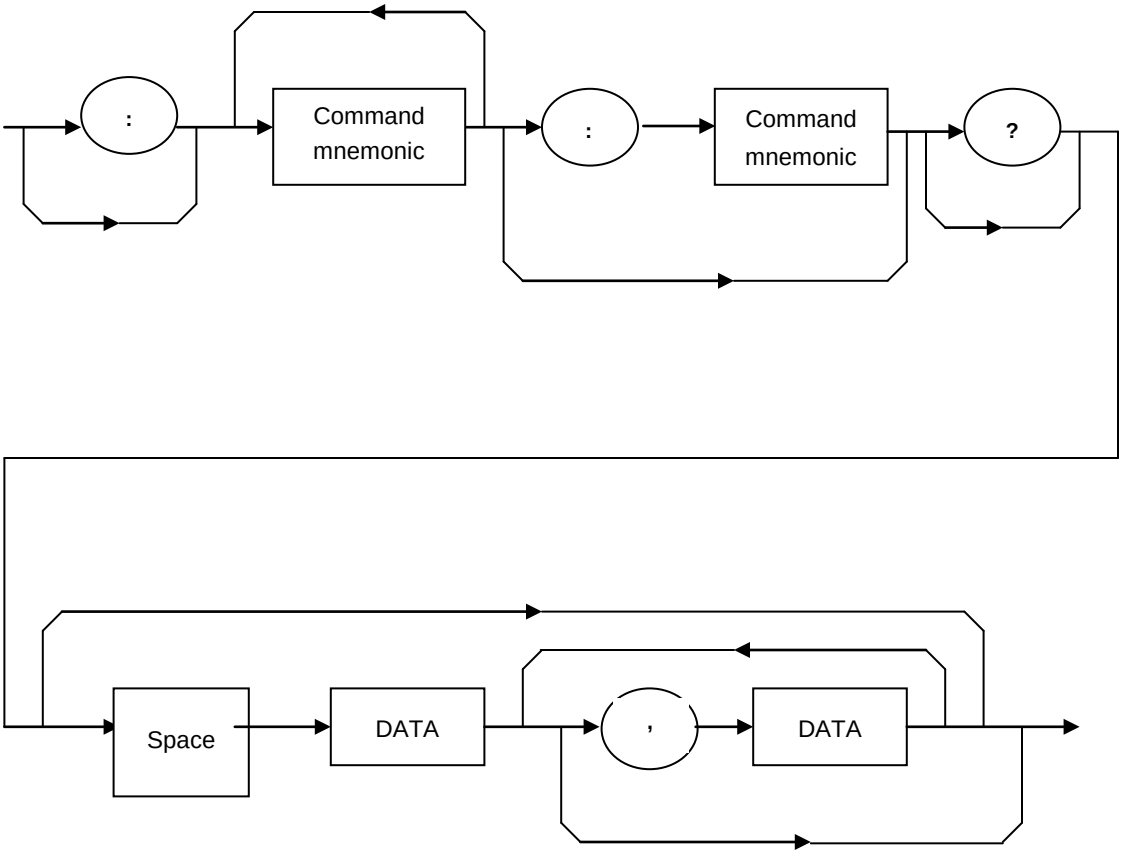
A brief description of the SCPI syntax is provided to allow basic operation. For a more detailed definition of SCPI the appropriate specification should be consulted.

SCPI Principles

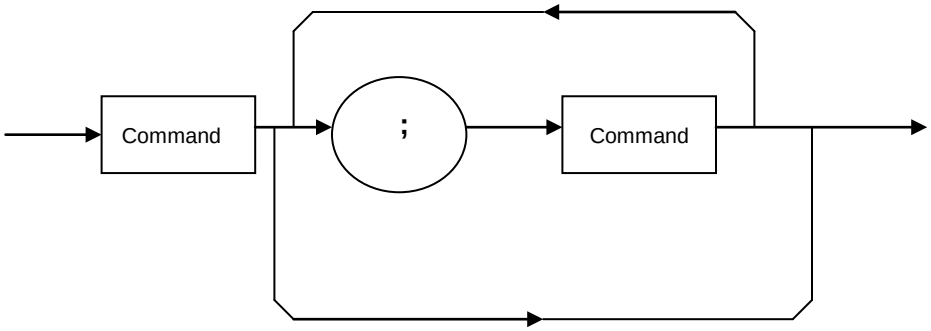
SCPI, in conjunction with IEEE 488.2, lays down rules for syntax and protocols for instrument control. If these rules are violated then a defined error process occurs.

The basic SCPI protocol relies on the bus controller sending information to an instrument. An instrument will never send information to the controller unless requested, or command to so by the bus controller. The controller can illicit responses from the 2461 by sending a query command (a command containing the ? character).

Commands sent to the 2461 conform to SCPI guidelines. The SCPI command set is a tree structure. All SCPI commands are grouped into various sub-systems (SENSE, CONFIGURE, MEASURE, INPUT, OUTPUT, CALCULATE, CALIBRATE, SOURCE, TEST, INSTRUMENT, STATUS and SYSTEM etc). The basic form of a SCPI command is:



Commands may be concatenated on the same line:



Commands and string data in SCPI have two forms, long form and short form, in the following command definitions and examples capitals are used to represent the short form and a mix of capitals and lowercase make up the long form. The 2461 can only accept the long or the short form as defined. Using a form between short and long form is not permissible.

Sending and Receiving SCPI Commands Over the VXI*bus*

SCPI commands are sent over the VXI*bus* by using the word serial command, *byte available.* See definition later. The last byte sent in a command line must either have the END bit set or a Line Feed must be appended to the end of the command line. The system controller will usually set the END bit for the user. It is permissible to terminate a command line with both an END bit set and a Line Feed appended provided the END bit is set when sending the Line Feed.

To receive a message from the 2461, the word serial command *byte request* command should be used. The previous message sent must have contained at least one query command. For each *byte request* command, a single word is read by the controller containing one byte of reply data. This repeats until the end bit in the response from the 2461 is set.

Selecting an Application

The 2461 general command set is available at all times. This includes the commands to interrogate which cards are installed and to select which card is to be addressed next. To determine which application cards are installed, the operator should send:

```
INST:CAT?
```

If two UCT application cards are fitted the reply will be:

```
COUNter1, COUNter2
```

To select counter 2 (card located at the lower end of the front panel) the operator should send:

```
INST:SEL COUNter2
```

The command set for card 2 (a counter) is now enabled.

Status and Error Reporting

Status and error reporting commands fall within the general command set so these can be accessed at any time. Status, and set up of the status system, is done by the STATUS subsystem while error messages are retrieved by the SYSTem:ERRor? Command. Further status is controlled and reported back by the IEEE 488.2 commands *ESR? , *ESE , *SRE and *STB?

System Timeout

In SCPI protocol, a query command generates a response which must be read back. In some circumstances a query command will fail and no response will be available. If this is the case, an attempt to read back the information queried can cause the system to appear to 'hang'. This is especially true of measurement type instruments such as UCTs where no signal means that a measurement cannot take place when the MEAS? Or READ? Is sent. To overcome this problem, the 2461 implements the SCPI command SYSTem:TIMeout command where the 2461 imposes a timeout value on certain query commands. Should an application not respond within the timeout value, the 2461 will generate a timeout error and produce an extremely large result to avoid a hang-up situation.

Setting the timeout value to zero disables the timeout function.

Note that the timeout value applies to all 2461 query commands including application commands.

Other timeout mechanisms may exist in the system controller and associated software and if the operator wishes to use the timeout facility of the 2461 then these timeouts must not pre-empt the 2461 timeout.

General Commands

List of General SCPI Commands

INSTrument command summary

Command	Description
:INSTrument	
:CATalog?	Lists the present instruments
:SELect	Selects an instrument by name
:SELect?	Returns the name of the currently selected instrument

STATus command summary

Command	Description
:STATus	Path to control operation registers:
:OPERation	
[:EVENt]?	Reads the event register
:ENABle	Programs the enable register
:ENABle?	Reads the enable register
:CONDition?	Reads the condition register
:PRESet	Returns status registers to default states

SYSTem command summary

Command	Description
:SYSTem	
:PRESet	Sets the device to the default state.
:VERSion?	Returns the SCPI year-version.
:ERRor?	Queries the Error Queue.
:TIMeout	Sets a time-out period
:TIMeout?	Returns the time-out period

INSTrument Subsystem

The INSTrument subsystem provides a mechanism to identify and select the different instruments present on the VXI module.

INSTrument command summary

Command	Description
:INSTrument	
:CATalog?	Lists the present instruments
:SELect	Selects an instrument by name
:SELect?	Returns the name of the currently selected instrument

INSTrument:CATalog?

Description:

This query command returns a comma-separated list which contains the names of individual instruments present on the 2461 module.

Example:

"INST:CAT? ==> COUNter1" one function card fitted.

"INST:CAT? ==> COUNter1,COUNter2" two function cards fitted.

INSTrument[:SELect] <identifier>

Description:

This command selects a specified instrument.

The following SCPI commands will only control the selected instrument.

Only one instrument can be selected at any time. The selection of an instrument automatically deselects the other instrument.

Parameter:

<identifier>: a string specifying the instrument to be selected:

Model	List of Identifiers
One function card	**COUNter1**
Two function cards	**COUNter1** or **COUNter2**

The identifier list can be retrieved by the "INSTrument:CATalog?" command.

Initialization:

*RST COUNTer1

Example:

"INST:SEL COUNter1" Selects the COUNter1 instrument

"MEAS:FREQ?" Takes a frequency measurement (COUNTer1)

"INST:CAT? ==> COUNter1,COUNter2" - Returns the identifier list of the module

"INST:SEL COUNter1" Selects the COUNter1 instrument

"INIT" Initiates a measurement (COUNter1)

"INST:SEL COUNter2" Selects the COUNter2 instrument

"INIT" Initiates a measurement (COUNter2)

Interrogative form:

Queries the currently selected instrument

INSTrument[:SELect] ? ==> <COUNter1 | COUNter2>

STATus subsystem

STATus command summary

Command	Description
:STATus	Path to control operation registers:
:OPERation	
[:EVENt]?	Reads the event register
:ENABle	Programs the enable register
:ENABle?	Reads the enable register
:CONDition?	Reads the condition register
:PRESet	Returns status registers to default states

General Description

- The IEEE-488.2 standard requires two registers: Status Byte and Standard Event Status Register.
- The Status Byte provides global system status information that the user may select through the Enable Registers.
- The Standard Event Status Register reports standard events such as Error Detection or Operation Complete.
- SCPI has added a few supplementary registers which represent the status and events specific to the instrument. Three registers are available at the same level:

A **CONDition** register reports the state of the instrument, and each state change for the instrument is indicated in the Condition register.

An **EVENt** register captures changes in the associated Condition register. When a change has been captured, the Event register signals the change, even if the Condition register has returned to its initial state.

A few bits of the Event register do not reflect changes in the corresponding Condition register. These bits represent the state of the selected bits in other Event registers. The selection of these bits is specified in the **ENABle** register.

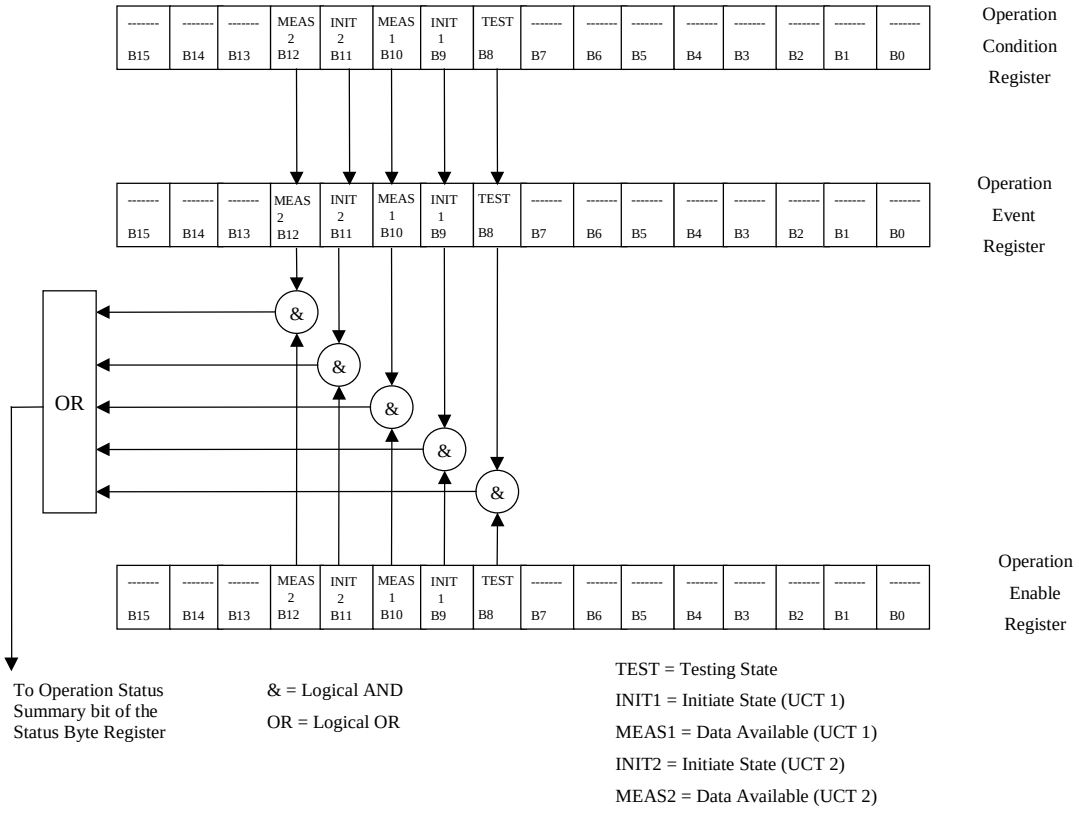

Operation Condition Register
Operation Event Register
Operation Enable Register
B15 B14 B13 B12 B11 B10 B9 B8 B7 B6 B5 B4 B3 B2 B1 B0
MEAS 2
INIT 2
MEAS 1
INIT 1
TEST
OR
&
To Operation Status Summary bit of the Status Byte Register
& = Logical AND
OR = Logical OR
TEST = Testing State
INIT1 = Initiate State (UCT 1)
MEAS1 = Data Available (UCT 1)
INIT2 = Initiate State (UCT 2)
MEAS2 = Data Available (UCT 2)

Figure 4-1, Operation Status Register.

STATus:PRESet

Description:

Clears to zero all bits of the Status Operation Enable Register.

Clears to zero all bits of the Status Operation Condition Register.

This command does not erase the EVENt registers.

****

STATus:OPERation[:EVENt]? ==> <numeric_value>

Description:

Returns the contents of the Operation Event register. Erases the contents of the Operation Event register after reading.

Refer to **Figure 4-2**.

****

STATus:OPERation:ENABle <numeric_value>

Description:

Sets the specified ENABle register. Each enable register is used as a mask for events. When a bit in an event enable register is cleared (0), the corresponding bit in the event register is masked and thus, cannot set the corresponding bit of the next register set in the status structure. Conversely, when a bit in an event enable register is set (1), the corresponding bit in the event register is unmasked. When the unmasked bit in the event register sets, the summary bit of the next register set in the status structure will set.

Refer to **Figure 4-2**.

Parameter:

<numeric_value> = 0 Clear enable register
= 256 Set bit B8 of enable register
= 512 Set bit B9 of enable register
= 1024 Set bit B10 of enable register
= 2048 Set bit B11 of enable register
= 4096 Set bit B12 of enable register

The sum of the decimal weights of the bits that you wish to set is sent as the

parameter (<numeric_value>) for the appropriate :ENABle: command.

Initialization:

*RST, *CLS	Not affected
Power-up:	0

Note that after power-up, by default, none of the bits of the STATus:OPERation:EVENt register are reported to bit 7 of the Status Byte.

Examples:

"STAT:OPER:ENAB 512" enables the INIT bit of the *STATus:OPERation:EVENt* to be reported in bit B7 of *the Status Byte.*

Interrogative form:

STATus:OPERation:ENABle? ==> <numeric_value>

****

STATus:OPERation:CONDition? ==> <numeric_value>

Description:

Returns the contents of the Operation CONDition register.

Refer to **Figure 4-2**.

Content of the Status Operation Condition Register:

Bit B8, Testing state (Test): Set when the instrument has completed the self-test process and clear when it starts the testing process.

Bit B9, Initiate state (Init1): Set when the first instrument is in the Initiate state (Trigger mode) and clear when it returns in the idle state (normal mode).

Bit B10, Data available (Meas1): Set when all of the required measurements have been completed by instrument 1. Cleared when the measurement is in progress.

Bit B11, Initiate state (Init2): Set when the second instrument is in the Initiate state (Trigger mode) and clear when it returns in the idle state (normal mode).

Bit B12, Data available (Meas2): Set when all of the required measurements have been completed by instrument 2. Cleared when the measurement is in progress.

Initialization:

*RST	Not affected
*CLS or STAtus:PRESet	0
Power-up:	0

SYSTem subsystem

SYSTem command summary

Command	Description
:SYSTem	
:PRESet	Sets the device to the default state.
:VERSion?	Returns the SCPI year-version.
:ERRor?	Queries the Error Queue.
:TIMeout	Sets a time-out period
:TIMeout?	Returns the time-out period

SYSTem:PRESet

Description:

Reinitializes the entire module.

Identical to the "*RST" common command.

Appendix C lists the default conditions for this command.

No Interrogative form

****

SYSTem:VERSion? ==> <numeric_value>

Description:

Returns the SCPI year-version to which the instrument conforms.

1996.0

SYSTem:ERRor? ==> <data_string>

Description:

Returns an error message in the Error Output Queue, a FIFO queue. The oldest error message is sent first and error messages are removed from the queue once sent. When the queue is empty, the message "0, No error" is sent. If an error occurs after the queue is already full, the message "350, Queue overflow" is stored instead of the error. Any further errors are not stored.

****

SYSTem:TIMeout <integer>

Description:

Sets the maximum amount of time a query measurement command (FETCh?, READ?, MEAS?) is allowed to take.

If the query command has not completed within the specified time, the command is aborted and a TIMEOUT error is placed in the error queue.

Only one time-out can be programmed to control all the instrument boards

Parameters:

<integer>: Represents the value of the time-out in seconds. It can take value from 0 to 100,000

0: Disables the time-out

Initialization:

10 seconds

Examples:

The following example waits 10 seconds for a scalar-type UCT frequency measurement triggered by a TTL Trigger line:

"SYST:TIM 10"

"INST:SEL COUN1"

“CONF:FREQ”

"ARM:START:SOUR TTLTRG7"

"READ?"

<Waits a maximum of 10 seconds>

The following example waits 2 seconds for an array-type UCT frequency measurement to complete 5 readings:

"SYST:TIM 2"

"INST:SEL COUN1"

“MEAS:ARRAY:FREQ? 5”

<Waits maximum of 100 seconds>

<Only 2 of the 5 measurements are armed and completed>

<+00003.9862505E+00, +00003.9862505E+00,+9.90000000000E+37, +9.90000000000E+37, +9.90000000000E+37>

Interrogative form:

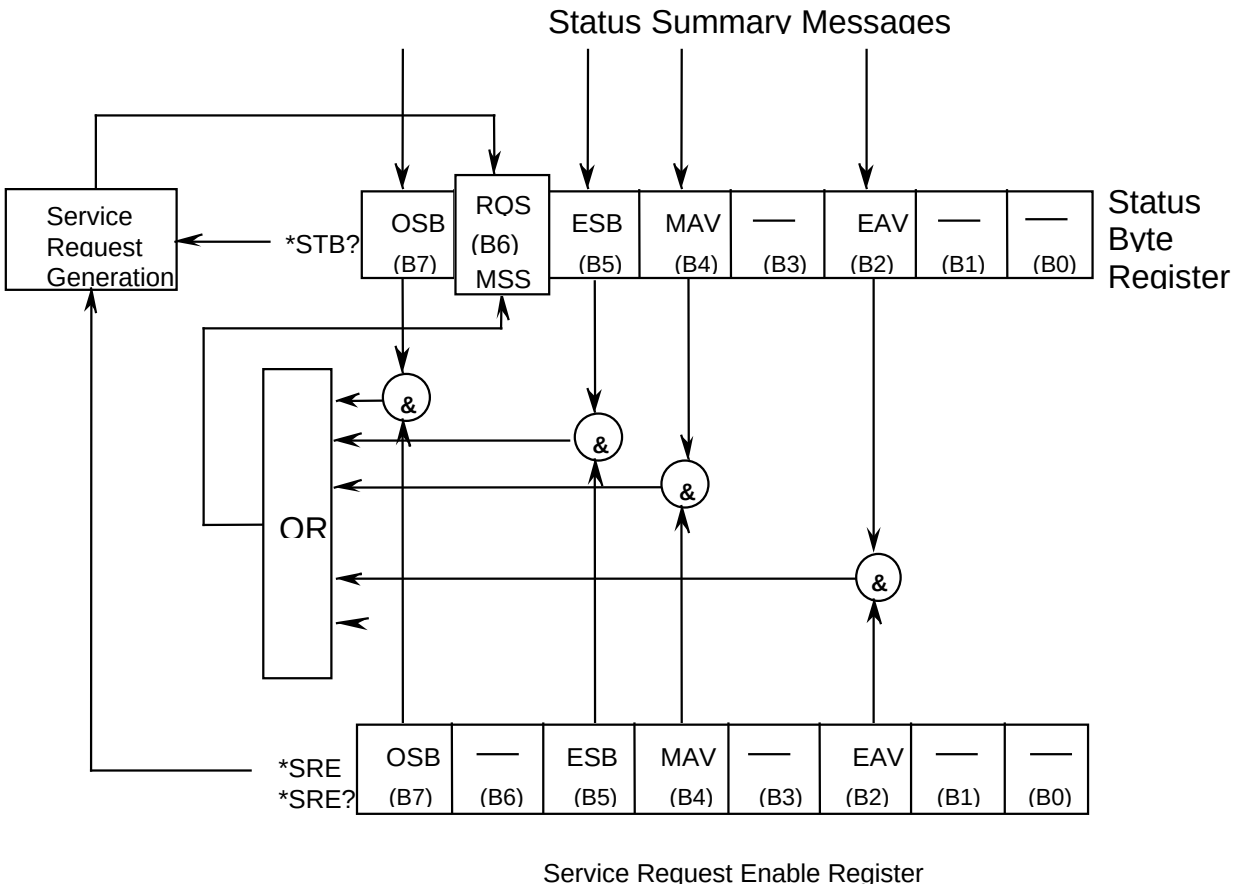
SYSTem:TIMeout?: <integer>

Queries the programmed time-out.

Common commands

IEEE 488.2 common command summary:

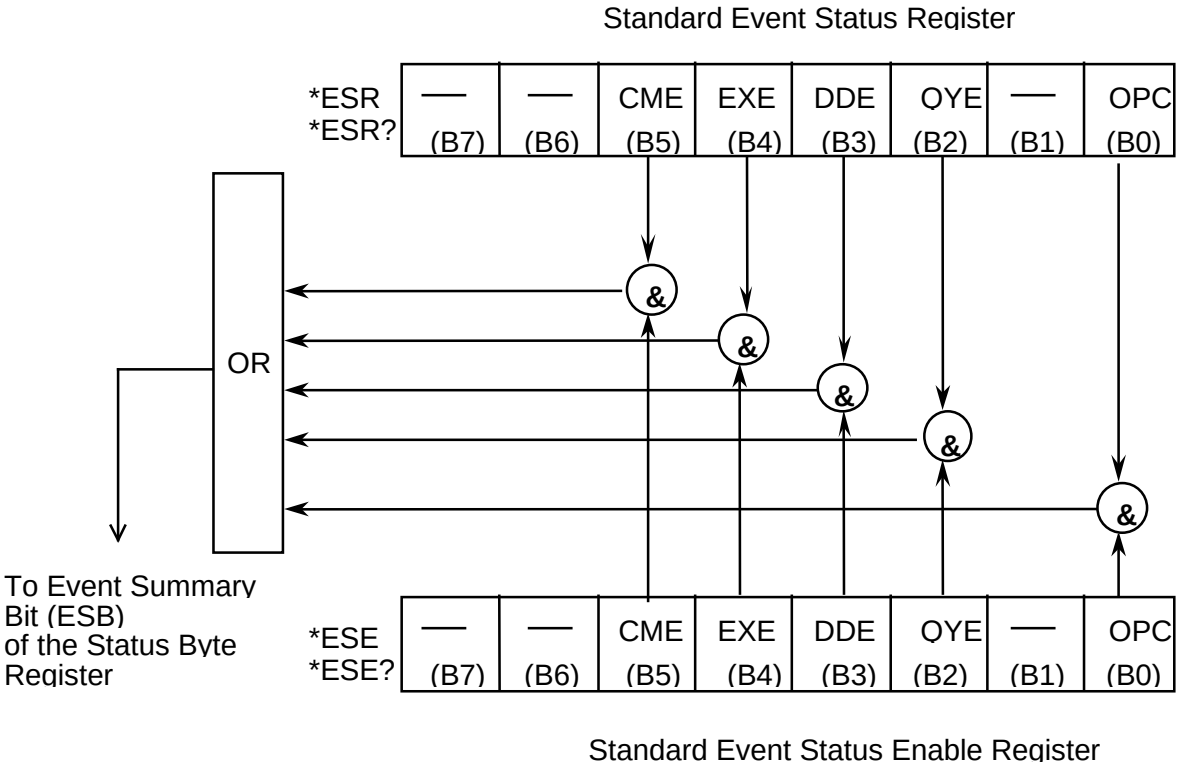
Command	Description
*CLS	Clears all Event register and Error queue.
*ESE	Sets the contents of the Standard Event Enable Register.
*ESE?	Request the programmed value of the Standard Event Enable Register.
*ESR?	Request the programmed value of the Standard Event Status Register and clears it
*IDN?	Request the manufacturer, model number, serial number and firmware revision of the module.
*OPC	Sets the Operation Complete bit in the Standard Event Status Register after all pending commands have been executed.
*OPC?	Places an ASCII ‘1’ into the VXI output buffer when all pending selected device operations have been completed.
*RST	Returns the module to the default conditions.
*SRE	Sets the contents of the Service Request Enable Register.
*SRE?	Returns the value of the Service Request Enable Register.
*STB?	Returns the value of the Status Byte Register.
*TST?	The module performs an autotest and returns the result.
*TRG	Causes the UCT to switch from the Idle to the Initiated state



OR = Logical OR
& = Logical AND

OSB = Operation Status Summary Bit
MSS = Master Summary Status
RQS = Request for Service
ESB = Event Summary Bit
MAV = Message Available
EAV = Error Available

Figure 4-2, Status Byte and Service Request



OR = Logical OR
& = Logical AND

CME = Command Error
EXE = Execution Error
DDE = Device-Dependent Error
QYE = Query Error
OPC = Operation Complete

Figure 4-3, Standard Event Status

*CLS

Description:

Erases all the Event registers, all Condition registers, the Error Queue and the Status Byte Register.

****

*ESE<numeric_value>

Description:

Sets the Standard Event Enable Register.

This command is sent with the decimal equivalent of the binary value that determines the desired state (0 or 1) of the bits in the register.

This register is used as a mask fort the Standard Event Status Register. When a standard event is masked (disables), the occurrence of that event will not set the Event Summary Bit (ESB) in the Status Byte Register. Conversely, when a standard event is unmasked (enabled), the occurrence of that event will set the ESB bit.

A cleared bit (bit set to 0) in the enable register will prevent (mask) the ESB bit in the Status Byte Register from setting when the standard event occurs. A set bit (bit set to 1) in the enable register will allow (enable) the ESB bit to set when the corresponding standard event occurs.

Refer to **Figure 4-3**.

Parameter:

<numeric_value>	= 0	Clear register.
	= 1	Set OPC (B0) of enable register
	= 4	Set QYE (B2) of enable register
	= 8	Set DDE (B3) of enable register
	= 16	Set EXE (B4) of enable register
	= 32	Set CME (B5) of enable register

The sum of the decimal weights of the bits that you wish to be set is the value that is sent with the *ESE command. For example, to set the CME and QYE bits of the Standard Event Enable Register, send the following command:

*ESE 36

where CME (bit B5) = Decimal 32

QYE (bit B2) = Decimal 4

parameter = 36

Initialization:

Power-up	Clears register
*CLS	No effect
*RST	No effect
STATus:PRESet	No effect

****

***ESE? ==> <numeric_value>**

Description:

Returns the programmed value of the Standard Event Enable Register.

This command does not clear the Standard Event Status Enable Register.

Refer to **Figure 4-3**.

***ESR? ==> <numeric_value>**

Description:

Returns the programmed value of the Standard Event Status Register.

A set bit in this register indicates that a particular event has occurred. For example, for an acquired decimal value of 48, the binary equivalent is 00110000. For the binary value, bits B4 and B5 of the Standard Event Status Register are set. These set bits indicate that a device-dependent error and a command error have occurred.

This command clears the Standard Event Status Register.

Refer to **Figure 4-3**.

Initialization:

Power-up	Clears register
*CLS	Clears register
*RST	No effect
STATus:PRESet	No effect

****

*IDN? ==> RACAL,<Model_number>,<Serial_Number>,<Firmware_revision> [-<FunctionCard1 Firmware_revision >[-<FunctionCard2 Firmware_revision>]]

Description:

Returns the model identification: manufacturer, model number, serial number and firmware revision(s).

For example:

RACAL,2461,U971301,01.03-01.03-01.03

The firmware revision field includes the 2461 revision (first number) and the Function Card Firmware Revision(s).

*OPC

Description:

After the *OPC command is sent, The Operation Complete bit (B0) of the Standard Event Status Register will set immediately after the last pending command is completed. If the corresponding bit (B0) in the Standard Event Enable Register and bit 5 (ESB) of the Service Request Enable Register (refer to *SRE command) are set, the RQS/MMS bit in the Status Byte Register will set.

Refer to **Figure 4-2 and 4-3**.

****

***OPC? ==> 1**

Description:

When this common command is sent, an ASCII '1' will be placed in the VXI Output buffer after the last pending operation is completed.

The '1' in the Output buffer will set the MAV (Message Available) bit (B4) of the Status Byte Register. If the corresponding bit (B4) in the Service Request Enable Register (refer to *SRE command) is set, the RQS/MMS bit in the Status Byte Register will set.

Refer to **Figures 4-2 and 4-3**.

****

***RST**

Description:

Reset the module.

Identical to the SCPI "SYSTem:PRESet" command.

Appendix C lists the default conditions for this command.

***SRE <numeric_value>**

Description:

Sets the contents of the Service Request Enable Register. This enable register is used along with the Status Byte Register to generate service request (SRQ). With a bit in the SRE register set, an SRQ will occur when the corresponding bit in the Status Byte Register is set by an appropriate event.

Refer to **Figure 4-2**.

Parameter:

<numeric_value>	=	0	Clears the register
		4	Set EAV bit (B2)
		16	Set MAV bit (B4)
		32	Set ESB bit (B5)
		128	Set OSB bit (B7)

The sum of the decimal weight of the bits that you which to set is the value that is sent with the *SRE command. For example, to set the ESB and MAV bits of the SRE register, send the following command:

*SRE 48

where ESB (bit B5) = Decimal 32
MAV (bit B4) = Decimal 16
parameter = 48

****

***SRE? ==><numeric_value>**

Description:

Read the contents of the Service Request Enable Register

***STB? ==> <numeric_value>**

Description:

Read the contents of the Status Byte Register.

A description of this register is shown in **Figure 4-2**. The EAV, MAV, ESB and OSB bits are set by other event registers and queues. Bit 6 (SRQ/MSS) sets when one or more enabled conditions occur.

The *STB? command does not clear the Status Byte Register. This register can only be cleared by clearing the related registers and queues

****

***TST? ==> <result>**

Description:

Execute a self-test on the module and returns the result. This command connects the standard frequency in use to the counting input and takes frequency measurements. Both UCT boards are tested. The test fails if the frequency measurement differs from 10MHz.

Result:	= 0	Tests passed
	= 1	Self-test failure, application board 1
	= 2	Self-test Failure, application board 2
	= 3	Self-test Failure, application board 1 and 2

Examples:

"*TST?" ==> 3: Self-test Failure in application board 1 and 2.

"*TST?" ==> 0: Tests passed.

"*TST?" ==> 1: Self-test Failure in application board 1.

***TRG**

Description:

It causes the UCT to switch from the Idle to the Initiated state.

If the Manual Totalize function is currently selected the *TRG command switches the state of the Manual Totalize, i.e. it is started if the state was off, or it is stopped if the state was on.

Example:

"*TRG": Forces the UCT to the Initiated state.

Error Messages

No Errors

Error Description [Description/Explanation/Examples]

Number

0 No Errors

[The queue is empty. Each error/event in the queue has been read or the queue was erased by power on].

Command Errors

Error Description [Description/Explanation/Examples]

Number

-101 Invalid Character

[An element of the syntax contains a character invalid for this type]

-102 Syntax Error

[Command not recognized, or incorrect data type found. For example, a character string was received when the instrument expected something else.]

-108 Illegal Parameter

[The number of parameters is greater than expected. For example, the *SRE command only accepts one parameter, so the command *SRE 180,34 is illegal.]

-109 Missing Parameter

[The number of parameters is less than required. For example, the *SRE command requires a parameter, so the command *SRE is illegal.]

-112 Command Mnemonic Too Long

[The command mnemonic contains at most twelve characters (see IEEE 488.2, 7.6.1.4.1).]

-113 Unknown Command

[The command syntax is correct, but the command is not defined for this instrument. For example, *XYZ is defined for no instruments.]

-120 Numeric Data Error

[Like Error -121, this error is generated when data appears that includes non-decimal numeric data types.

This error message may appear if the instrument cannot detect a more specific error.]

or

Invalid Data character
[An invalid character has been received in a <data_string>.]

-121 Invalid Character in Number

[An invalid data type character has been received. For example, an alphabetic character in a number.]

-144 Data String Too Long

[The <data_string> element contains more than twelve characters (see IEEE 488.2, 7.7.1.4).]

Execution Errors

Error Description [Description/Explanation/Examples]

Number

-200 Execution Error

[This error occurs when the VXI output buffer is full, or if the INTERRUPT QUEUE is full, or if the calibration date has been forgotten when closing the calibration process.]

-210 Trigger Error

-211 Trigger Ignored

[A *TRG or trigger signal has been received and recognized by the instrument but, for timing considerations, has been ignored. For example, the instrument is not ready to respond.]

-221 Configuration Conflict

[A command has been interpreted but cannot be executed because of the current state of the instrument.]

Examples:
[While the device is in INITiate state, some commands are not allowed.
A TTLTrg line is already forced in input or output.
An other TTLTrg line is already enabled in Output.
A CALCulation SCPI command is not allowed while Calculation state is OFF.]

-222 Data Out of Bounds

[A command has been interpreted but cannot be executed because its value is out of bounds.]

-224 Illegal Parameter Value

[Used when a certain value, when possible, is expected. For example, the “CAL:STAT HOLD” command generates an Error because the interpreter expects an ON or OFF parameter instead of HOLD.]

-230 Data Corrupt or Stale

[A FETCh? command is sent but no data is available while the device is in idle state.]

-240 Hardware Error

[A command or request may not be executed because it is not allowed. For example, no instrument is selected and the user sends a MEASure command.]
or
[A time-out occurs.]

-241 Hardware Missing

[A command or request may not be executed because the corresponding hardware does not exist. For example, an option is not installed.]

Instrument-Specific Errors

Error Description [Description/Explanation/Examples]

Number

-311 Memory-Related Error

[An error has occurred in one of the memory areas of the module. The problem may arise when programming the Non-volatile memory or when transferring samples in DMA mode.]

-350 Queue Overflow

[A specific code has been entered in the Queue instead of the normal error code, which caused the error. This code indicates that the Queue is full and that the error that occurred has not been recorded.]

VXI*bus* Word Serial commands

The 2400 VXI Instrumentation module supports the following commands. They are described in detail in the following paragraphs.

Abort Normal Operation
Assign Interrupt Line
Asynchronous Mode Control
Begin Normal Operation
Byte Available
Byte Request
Clear
Control Event
End Normal Operation
Read Interrupter Line
Read Interrupters
Read Protocol
Read Protocol Error
Read STB

Abort Normal Operation

This command is used to halt normal operations. When the instrument receives this command, it returns to its default configuration, aborting any ongoing operations. The aborted state is defined as follows:

Waiting interrupts are not acknowledged, new bus or interrupt requests may not be acknowledged. The instrument is in an inactive state and is ready to receive commands.

Command:

$C8FF_{16}$

The response is placed in the Data Low Register.

Response:

$FFFE_{16}$

Assign Interrupt Line

This command is used to assign an IRQ line of the VME bus to a slave interrupt generator.

Command:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	1	0	1	0	1	0	X	Int_ID			X	Line		

X: Insignificant

Int_ID: Unique identifier of the assigned interrupt generator. For this model, the 'INT_ID' field has the value '001'.

Line: Number of the IRQ line on the VME bus. A zero value indicates that the interrupt generator is disconnected.

Action:

if Int_ID = 001, save the value of 'Line'

The response is placed in the Data Low Register.

Response:

if Int_ID = 001 --> $FFFE_{16}$

else --> $7FFE_{16}$

Asynchronous Control Mode

A ‘master’ uses this command to create a path for events and responses.

Command:

15	14	13	12	11	10	9	8	7 6 5 4	3	2	1	0
1	0	1	0	0	0	1	0	X	Val Resp*	Val Event*	Resp Mode	Event Mode

X: Insignificant

Val Resp*: A zero (0) enables reply generation. A one (1) disables reply generation.

Val Event*: A zero (0) enables event generation. A one (1) disables event generation.

Resp. Mode: A one (1) indicates that responses can be sent as signals. A zero (0) indicates responses can be sent as interrupts.

Event Mode: A one (1) indicates that events can be sent as signals. A zero (0) indicates that events can be sent as interrupts.

The response is place in the Data Low Register:

Response:

15 14 13 12	11	10	9	8	7	6	5	4	3	2	1	0
Status	1	1	1	1	1	1	1	1	Val Resp*	Val Event*	Resp Mode	Event Mode

Status: This field indicates the status of command execution.

- F_{16}: The command has executed correctly.

- 7_{16}: The command did not execute. A required option was not supported.

Val Resp*: A zero (0) enables reply generation. A one (1) disables reply generation.

Val Event*: A zero (0) enables event generation. A one (1) disables event generation.

Resp.Mode: A one (1) indicates that responses can be sent as signals. A zero (0) indicates that responses can be sent as interrupts.

Event Mode: A one (1) indicates that events can be sent as signals. A zero (0) indicates that events can be sent as interrupts.

Begin Normal Operation

This command tells the instrument it can execute a normal operational cycle.

Command:

$FCFF_{16}$ or $FDFE_{16}$

The response is placed in the Data Low Register.

Response:

$FFFE_{16}$

Byte Available

A ‘master’ uses this command to send a byte of data to the ‘slave’. The END field indicates the last byte of the message.

Command:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	1	1	1	1	0	END	DATA							

Response:

No response.

Byte Request

The ‘master’ uses this command to receive a data byte from a ‘slave’ instrument.

Command:

$AEFF_{16}$

The response is placed in the Data Low Register. The END field indicates the last byte of the message.

Response:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	1	1	1	1	0	END	DATA							

Clear

This command is used by a ‘master’ to force a ‘slave’ to re-initialize the VXI interface and interrupt all operations. Any operation underway when this command is received is terminated normally.

Command:

$FFFF_{16}$

Response:

No response

Control Event

This command is used by a 'master' to selectively enable events generated by the 'slave'. A one (1) in the validation field enables specific event generation. A zero (0) in the validation field disables specific event generation.

Command:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	1	0	1	1	1	1	Enable	Event						

Event : These bits (6 <- 0) identify the bits (14 <- 8) of the enabled/disabled event. See Section E.4 of the VXI*bus* Specification, "Protocol Events", for more information.

The response is placed in the Data Low Register.

Response:

$FFFE_{16}$: The command terminated normally.
$7FFE_{16}$: Command default. The instrument does not generate the referenced event.

End Normal Operation

This command is used by the 'master' to methodically halt all operations. Upon receipt of this command, the 'master" sends an End Normal Operation command to all message-based 'slaves'. The 'master' is responsible for halting the 'slaves'.

The 'End' state is defined as follows: waiting interrupts are not acknowledged, new bus or interrupt requests may not be acknowledged. The instrument is in an inactive state, and is ready to receive other commands.

Command:

$C9FF_{16}$

The response is placed in the Data Low Register.

Response:

$FFFE_{16}$

Read Interrupt Line

This command is used to determine to which of the VME bus IRQ lines a slave's particular interrupt generator is connected.

Command:

15	14	13	12	11	10	9	8	7 6 5 4 3	2 1 0
1	0	0	0	1	1	0	1	X	Int_ID

X: Insignificant

Int_ID: Unique identifier of the specific interrupt generator. For this module the 'Int_ID' field must contain the value '001'.

The VME bus' IRQ line number is placed in the Data Low Register.

Response:

15 14 13 12	11	10	9	8	7	6	5	4	3	2 1 0
Status	1	1	1	1	1	1	1	1	1	Line

Status: This field indicates the status of command execution.
- F_{16}: The command has executed correctly.
- 7_{16}: Command default. The interrupt generator referenced in the Int_ID field is unknown to the instrument.

Line: The IRQ line number on the VME bus. A value of zero (0) indicates the interrupt generator is disconnected.

Read Interrupts

This command is used to determine the number of a ‘slave’s’ interrupt generator.

Command:

$CAFF_{16}$

The number of the interrupt generator is placed in the Data Low Register.

Response:

$FFF9_{16}$ (an interrupt generator).

Read Protocol

A ‘master’ uses this command to determine what protocol is used in the word serial protocol used by the ‘slave’.

Command:

$DFFF_{16}$

The supported protocol word is placed in the Data Low Register.

Response:

$FE23_{16}$

The 2400 VXI Instrumentation:

-- is capable of generating the events.

-- supports the Read Interrupts, Read Interrupt Line, and Assign Interrupt Line commands.

-- supports the Word Serial Trigger commands.

-- supports the VXI IEEE-488.2 Instrument protocol.

-- supports the VXI Instrument protocol..

Read Protocol Error

A 'master' uses this command to ask its 'slave' to report its current error status in the form of a 16-bit response. After having replied to this command, the 'slave' resets its Current Error status to 'No Error'.

Command:

$CDFF_{16}$

The error code is placed in the Data Low Register.
There are many possible responses:

No Error: $FFFF_{16}$

Multiple Queries: $FFFD_{16}$

The instrument was instructed to write over a previous unread response.

Unsupported command: $FFFC_{16}$

The instrument received an unsupported command.

DIR violation: $FFFB_{16}$

The instrument received a command that violates the DIR protocol.

DOR violation: $FFFA_{16}$

The instrument received a command that violates the DOR protocol.

Read Ready violation: $FFF9_{16}$

The instrument received a command that violates the Read Ready protocol.

Write Ready violation: $FFF8_{16}$

The instrument received a command that violates the Write Ready protocol.

Read STB

This command allows a ‘master’ to read the status byte of a ‘slave’.

Command:

$CFFF_{16}$

The status byte is placed in the Data Low Register.

Response:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	1	Status Byte							